CS3700 - Introduction to Database Systems
Assignment-3: SQL on RealEstate DB

R.Yeshaswini CS22B001

Query 1

Finds the names and email addresses of buyers who have shown interest in properties located in the city of Chennai.

```sql
SELECT DISTINCT B.Name, BE.Email
FROM buyer B
JOIN potentialbuyers PB ON B.BuyerID = PB.BuyerID
JOIN property P ON PB.PropertyID = P.PropertyID
JOIN buyeremail BE ON B.BuyerID = BE.BuyerID
WHERE P.City = 'Chennai';
```

Query 2

Finds cities where properties were sold more than once and calculates how much property prices have increased over time.

$$\text{Price\_growth\_percentage} = \left(\frac{\text{Latest\_Price} - \text{Earliest\_Price}}{\text{Earliest\_Price}}\right) \times 100$$

```sql
SELECT
    p.City,
    MIN(t.TransactionDateTime) AS Earliest_Sale,
    MAX(t.TransactionDateTime) AS Latest_Sale,
    MIN(t.PropertyValue) AS Earliest_Price,
    MAX(t.PropertyValue) AS Latest_Price,
    ((MAX(t.PropertyValue) - MIN(t.PropertyValue)) * 100.0 / MIN(t.PropertyValue))
        AS Price_growth_percentage
FROM property p
JOIN transaction t ON p.PropertyID = t.PropertyID
GROUP BY p.City
HAVING COUNT(DISTINCT t.TransactionDateTime) > 1
ORDER BY Price_growth_percentage ;
```

Query 3

Finds the most popular type of large property (with size over 1000 sq ft) in each city
Popularity here means the number of large properties of that type.

```sql
SELECT P.City, P.Type, COUNT(*) AS Num_properties
FROM property P
WHERE P.Size > 1000 AND NOT EXISTS (
    SELECT 1
    FROM property P2
    WHERE P2.City = P.City AND P2.Size > 1000 AND P2.Type <> P.Type AND
    (
          SELECT COUNT(*)
          FROM property P3
          WHERE P3.City = P2.City AND P3.Type = P2.Type AND P3.Size > 1000
      ) > (
          SELECT COUNT(*)
          FROM property P4
          WHERE P4.City = P.City AND P4.Type = P.Type AND P4.Size > 1000
      )
)
GROUP BY P.City, P.Type;
```

Query 4

Finds each property owner and calculates the total revenue they earned from selling properties

```sql
SELECT O.OwnerID, O.Name AS OwnerName, SUM(T.PropertyValue) AS TotalRevenue
FROM owner O
JOIN property P ON O.OwnerID = P.OwnerID
JOIN transaction T ON P.PropertyID = T.PropertyID
GROUP BY O.OwnerID, O.Name
ORDER BY TotalRevenue ;
```

Query 5

Finds the most expensive property transaction in the database.

```sql
SELECT T.TransactionID, T.PropertyValue, T.TransactionDateTime, B.Name AS
    BuyerName, P.City, A.Name AS AgentName
FROM transaction T
JOIN buyer B ON T.BuyerID = B.BuyerID
JOIN property P ON T.PropertyID = P.PropertyID
JOIN agent A ON T.AgentID = A.AgentID
WHERE T.PropertyValue = (
    SELECT MAX(PropertyValue) FROM Transaction
);
```

Query 6

Find agents whose average sale price is higher than the global average sale price, and for each of them, also find Their least active city (where they sold the few properties), The total number of properties they sold, Their average sale price.

```sql
SELECT
    a.AgentID,
    a.Name,
    COUNT(t.TransactionID) AS total_sales,
    AVG(t.PropertyValue) AS avg_agent_sale_price,
    (
        SELECT ag2.City
        FROM propertyagent pa2
        JOIN property p2 ON pa2.PropertyID = p2.PropertyID
        JOIN agent a2 ON pa2.AgentID = a2.AgentID
        JOIN agency ag2 ON a2.AgencyID = ag2.AgencyID
        WHERE pa2.AgentID = a.AgentID
        GROUP BY ag2.City
        ORDER BY COUNT(*)
        LIMIT 1
    ) AS least_active_city
FROM agent a
JOIN propertyagent pa ON a.AgentID = pa.AgentID
JOIN property p ON pa.PropertyID = p.PropertyID
JOIN transaction t ON t.PropertyID = p.PropertyID
GROUP BY a.AgentID, a.Name
HAVING AVG(t.PropertyValue) > (
    SELECT AVG(PropertyValue) FROM transaction
);
```